Progress Report on the Path Planning of Webot Satisficing Experiment

Min Zheng

July 16, 2017

This project considers the integrated navigation and control of an unmanned ground vehicle(UGV) deployed to visually classify multiple targets in an obstacle-populated environment. This report describes the setup of the path planning problem and proposes some potential problems and plans.

I. WORKSPACE CONSTRUCTION

The workspace is the same as the one used for active satisficing human studies. The UGV and 30 target objects are in the same environment that consists of four rooms, denoted as the region-of-interest (ROI). Similar to the human studies, there are 30 targets, denoted as $\mathcal{T} = \{\mathcal{T}_i; i = 1, 2, ...30; \mathcal{T} \subset W\}$. The targets are considered as points. A two-dimensional representation from the webot environment was extracted, and the workspace is constructed in MatLab, as shown in the figure below (Figure 2). The construction of workspace is built in an object oriented manner with classes of obstacles, targets and robot, each has properties and functionalities.
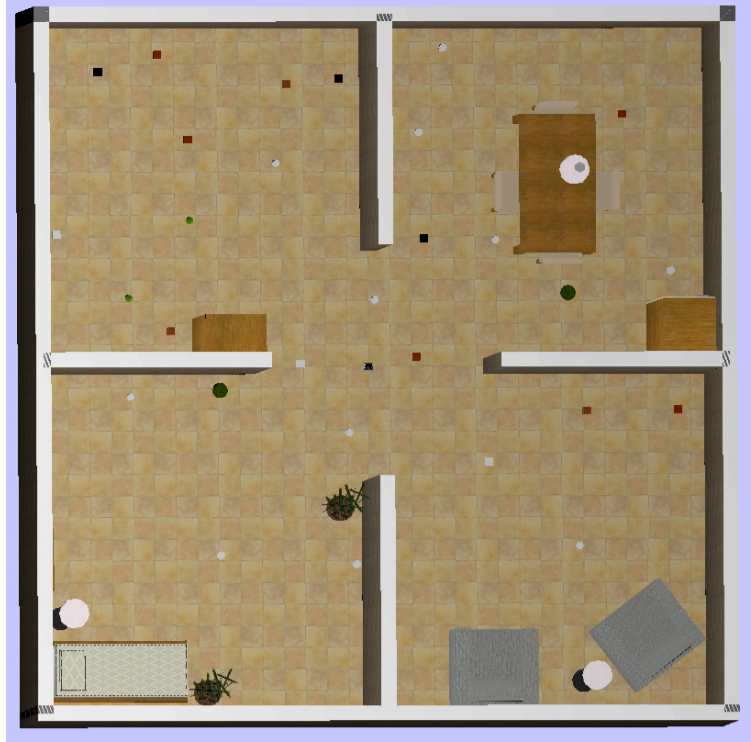
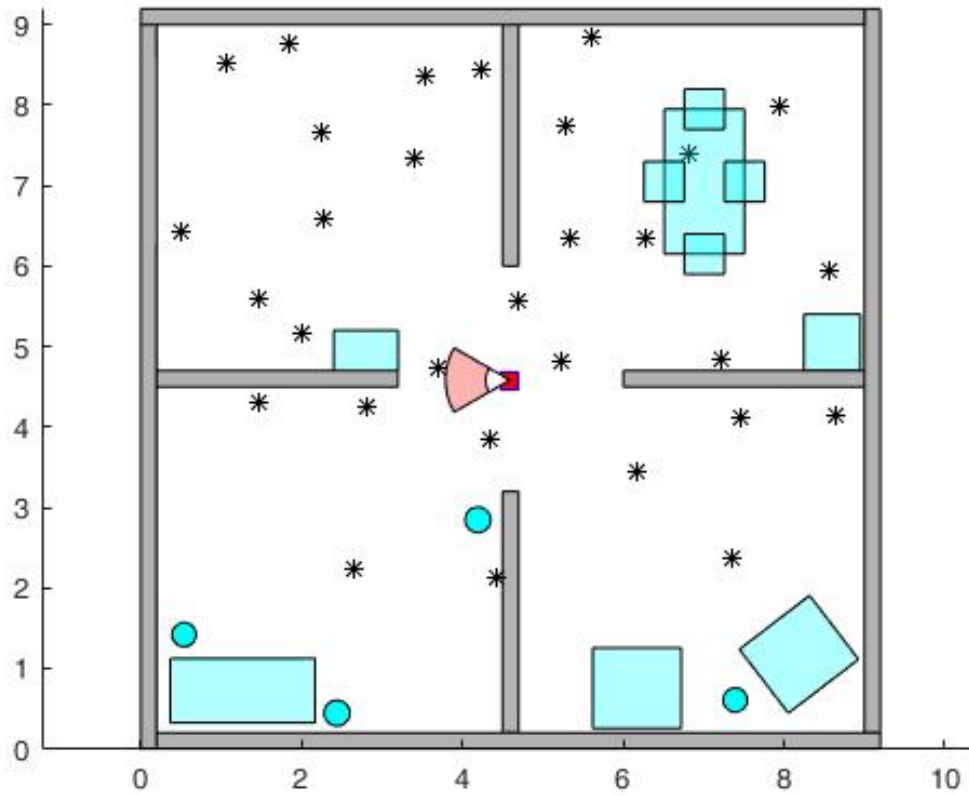Fig. 1: Top view of the webot environment

Fig. 2: 2D representation built in MatLab. Obstacles are represented by colored regions, and targets represented by the stars. The UGV is the red square, while its FOV is the red area within the fan shape.
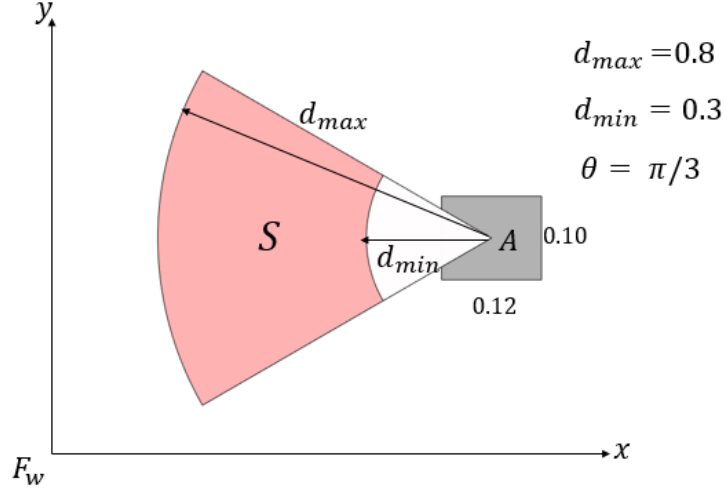
Fig. 3: FOV of the UGV when the camera is facing $-pi$

II. FOV AND C-TARGET GENERATION

The UGV sensor in the webot environment has platform geometry $\mathcal{A}$, and FOV geometry $S \subset \mathbb{R}^2$. The dimension of UGV is 0.12m x 0.10m in 2D. Its translation step is 0.01m, and rotation step is $\pi/12$.

In order to combine with Zeyu's visual classification project, the FOV was determined based on requirement of images taken. A successful classification requires the image to be taken within a minimum and a maximum distance, denoted by d_{min} and d_{max}. In human tests, the open angle of FOV for the on board camera is $\theta = \pi/3$. Therefore the same θ is used here. Also, in the visual classification problem, Ryan took all the pictures from 0.3m. If the picture is taken less than that distance, the target object is not guaranteed to be completely within the camera view. Meanwhile, in the human tests software setup, the robot must be within 1m from the target to measure or classify. In such manner, the range of distance from target is conservatively set within $d_{min} = 0.3m$ and $d_{max} = 0.8m$. In order to better work with the visual classification, these parameters, especially d_{max} may be subject to changes due to the highly sensitive manner of visual classifier. For example, it may be affected by whether the visual classifier allows multiple objects to appear in the view.

Assume that the UGV camera can freely rotate. C-Target is defined so that the target $\mathcal{T}_i$ in $\mathcal{W}$ maps in the robot's configuration space $\mathcal{C}_{free}$ to the C-target region $\mathcal{CT}_i = \{q \in \mathcal{C}_{free}|\mathcal{S}(q) \cap \mathcal{T}_i \neq$

$\emptyset\}$. The computation of C-target is illustrated for a single target in Figure 4. In (a), the robot is within C-target, since the target is in the robot's FOV when it faces $-pi/2$. In (b), the robot is not in C-target, and it can not detect the target no matter how it rotates at its current position.

Then, at every location in $\mathcal{C}_{free}$, as the robot camera rotates freely but no any target is in its FOV, then such location is excluded from C-target. The preliminary C-Target is plotted in the workspace (Figure 5). C-Target is denoted by white, where the rest region in $\mathcal{C}_{free}$ is grey. The implementation may need more modifications though. First, obstacles that may block the camera view need to be considered. For example, for the target denoted by red star in Figure 5, the robot view will be blocked by the wall, so such region should be removed from C-target. Secondly, for now, all of the obstacles are treated as not traversable. However, in the human tests, the UGV can go under the table and the chairs.

III. NEXT STEP

In the next step, the EER based on initial cues of the targets should be added, as well as a combination of uniform PDF and a Gaussian PDF. After the hybrid PDF is obtained, the milestone can be generated to construct the roadmap.
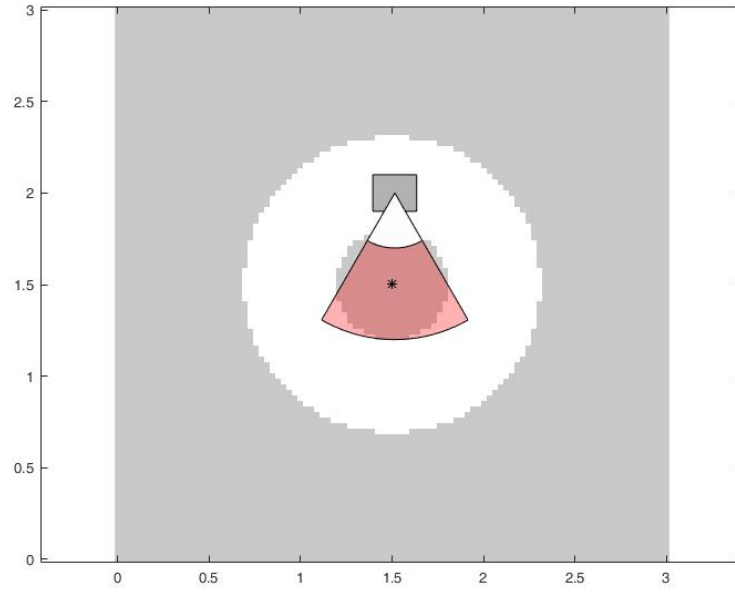
(a) robot is in C-target
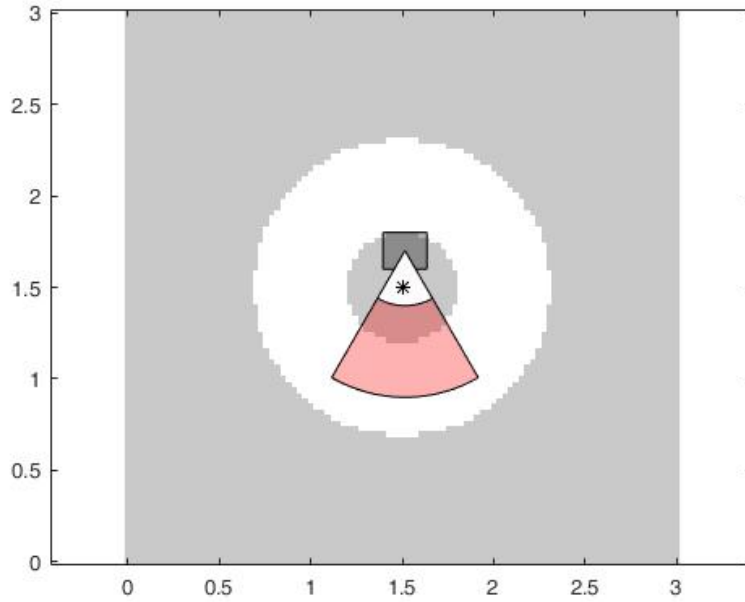
(b) robot is not in C-target

Fig. 4: Example of C-target with one target. C-target denoted by white area, and the star denotes the target. The target must fall in the red region(robot FOV) to be classified or measured.
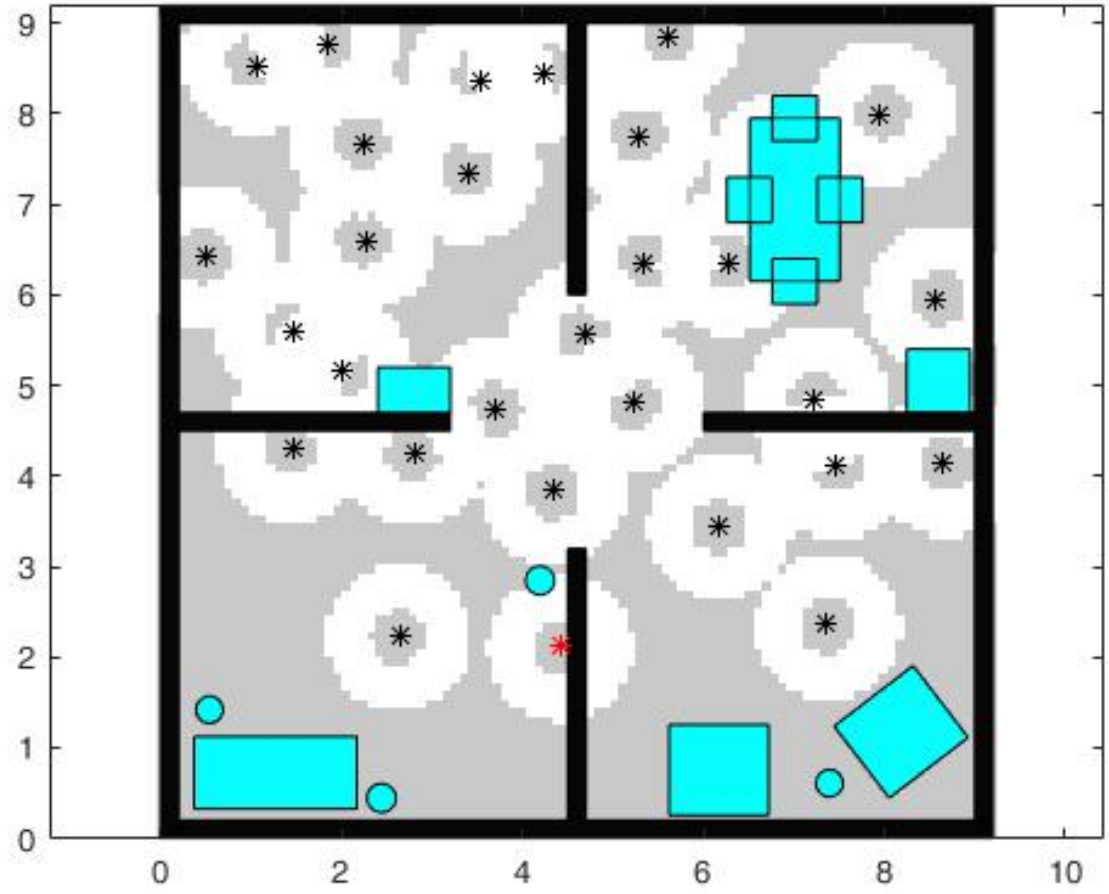
Fig. 5: C-Target is denoted by white, where the rest region in $\mathcal{C}_{free}$ is grey.